How to Implement GTID Replication in MySQL 5.6

FromDual GmbH
Annual company meeting 2014, Barcelona

Abdel-Mawla Gharieb
MySQL Support Engineer
abdel-mawla.gharieb@fromdual.com
GTID Replication

➢ What is GTID ?
➢ GTID benefits
➢ GTID Important Variables.
➢ GTID Replication Implementation (fresh installation).
➢ Adding new slave to GTID replication.
What is GTID ?
What is GTID?

GTID is a global transaction identifier which consists of two parts separated by a column:

{source_id:transaction_id}

source_id: server's UUID.

transaction_id: sequence number.

b9b4712a-df64-11e3-b391-60672090eb04:3
GTID Benefits
GTID Benefits

- Setting up MySQL replication is so simple now!
- Consistency is guaranteed between master and slaves.
- Simple to determine inconsistency
- Fail-over process become much easier.
- Automatic fail-over script is not a pain now.
GTID Important Variables
GTID Important Variables

- **gtid-mode**: ON|OFF
- **enforce-gtid-consistency**: prevent executing the non-transactionally safe statements.
- **gtid-purged**: transactions have been purged from the binary logs.
- **gtid-executed**: transactions already executed on the server.
- **gtid-next**: GTID for the next transaction.
GTID Replication Implementation
(Fresh installation)
GTID Replication Implementation

- Master's side configurations.
- Slave's side configurations.
Master's side configurations

- Add the following variables to my.cnf:

```ini
[mysqld]
server-id = 1
log-bin = mysql-bin
binlog_format = ROW
gtid_mode = on
enforce_gtid_consistency
log_slave_updates
```
Master's side configurations

- Restart MySQL to apply the configuration changes:

  shell> service mysql restart

- Create a MySQL user to be used by the slave:

  SQL> GRANT REPLICATION SLAVE ON *.* TO 'slave_user_name'@'slave_ip' IDENTIFIED BY 's3cret';
Slave's side configurations

- Add the following variables to my.cnf:

```ini
[mysqld]
server_id = 2
log_bin = mysql-bin
binlog_format = ROW
skip_slave_start
gtid_mode = on
enforce_gtid_consistency
log_slave_updates
```
Slave's side configurations

- Restart MySQL to apply the configuration changes:

  shell> service mysql restart

- Execute the CHANGE MASTER TO command:

  SQL> CHANGE MASTER TO
      MASTER_HOST='master_ip',
      MASTER_PORT=3306,
      MASTER_USER='slave_user_name',
      MASTER_PASSWORD='s3cret',
      MASTER_AUTO_POSITION=1;
Slave's side configurations

- Start the replication:

  SQL> START SLAVE;
Checking the replication!!

- Check the replication status:

```
SQL> SHOW SLAVE STATUS
Slave_IO_State: Waiting for master to send event
   Master_Host: 127.0.0.1
   Master_User: gtid_repl
   Master_Port: 3320

   Slave_IO_Running: Yes
   Slave_SQL_Running: Yes

Retrieved_Gtid_Set: b9b4712a-df64-11e3-b391-60672090eb04:1-2
Executed_Gtid_Set: b9b4712a-df64-11e3-b391-60672090eb04:1-2
   Auto_Position: 1
```
Adding new slave to a GTID Replication
Adding new slave to GTID Replication

- Backup the master server:

  ```
  shell> mysqldump -u root -p --all-databases --flush-privileges --single-transaction --flush-logs --triggers --routines --events --hex-blob > /path/to/backupdir/full_backup-$TIMESTAMP.sql
  ```

- Modify the new slave's my.cnf as described above.
- Restore the master backup file on the slave.
- Use change master to with `MASTER_AUTO_POSITION=1`
- Start the slave.
Adding new slave to GTID Replication

- mysqldump knows about GTID !!

```
--
-- GTID state at the beginning of the backup
--
SET @@GLOBAL.GTID_PURGED='b9b4712a-df64-11e3-b391-60672090eb04:1-7';
```

www.fromdual.com
We have time for some face-to-face talks...

- FromDual provides neutral and independent:
  - Consulting
  - Remote-DBA
  - Support for MySQL, Galera, Percona Server and MariaDB
  - Training

www.fromdual.com/presentations